Subject: A function to locate values... without using histogram
Posted by penteado on Thu, 16 May 2013 21:11:34 GMT
View Forum Message <> Reply to Message

I am writing this just to share on the solution I wrote to a problem I encounter frequently: Locating all occurrences of each different value contained in an array. Often this happens when I am working with database-like operations, when I have tables (arrays of structures) and need to do what would be the equivalent of several call to where():

```
foreach value,values do begin
  w=where(array eq value)
  (do stuff with w)
endforeach
```

Which is inneficient, due to the repeated searches over the entire array. So I would usually do this with a single call to histogram() and use its reverse indices. But when the array values are not sequential integers (I often need to do this with strings and doubles), they have to be converted to sequential integers, using uniq(). What I realized then is that uniq(), alone, does all the work:

```
s=sort(array)
sarray=array[s]
sr=(ul64indgen(n_elements(array)))[s]
;sr maps sarray back into array: sarray=array[s] and array=sarray[sr]

uinds=uniq(sarray)
uarray=sarray[uinds]
ret=hash()
last=0ULL
foreach el,uinds,i do begin
  nels=el-last+1
  els=sr[last+ul64indgen(nels)]
  ret.set,uarray[i],els
  last=el+1
endforeach
```

So I wrote a function to do this, pp_locate(), which I can use as

```
 array=['a','j','kk','a','a','b','zrdc','29','b','29','-19',' 0']
 loc=pp_locate(array,sorted_values=sarray,unique_values=uarra y,histogram=h)
help,loc
;LOC HASH <ID=140 NELEMENTS=8>
print,loc
;zrdc: 6
;a: 0 3 4
```

;j: 1
;0: 11
;-19: 10
;b: 5 8
;kk: 2
;29: 7 9

It can be found at

http://www.ppenteado.net/idl/pp_lib/doc/pp_locate.html
